What is claimed is:

1. An apparatus comprising:

a decode circuit to decode a load fence instruction;

an execution unit to execute said load fence instruction after said decode circuit has decoded said load fence instruction.

2. The apparatus of claim 1 wherein said execution unit prevents load operations from executing until load operations executed prior to said load fence instruction are globally observed.

3 The apparatus of claim 1 further comprising:

a first storage location to store at least a first bit to enable a pre-serialization mode;

a second storage location to store at least a second bit to enable a post-serialization mode.

4. The apparatus of claim 3 wherein said pre-serialization mode, when enabled, causes said execution unit to execute all loads prior to a load fence instruction and said post-serialization mode, when enabled, causes said execution unit to prevent loads after a load fence instruction from being executed.

5. The apparatus of claim 1 further comprising a control register to control enabling of a pre-serialization mode, a post-serialization mode, and an enhanced mode, wherein said pre-serialization mode, when enabled, causes said execution unit to execute all loads prior to a load fence instruction, said post-serialization mode,

when enabled, causes said execution unit to prevent loads after a load fence instruction from being executed, and said enhanced mode causes said execution unit to allow execution of operations other than load operations that are subsequent to a load fence instruction.

6. The apparatus of claim 4 wherein said execution unit allows an instruction subsequent to said load fence instruction to execute out of order with respect to said load fence instruction if there is no dependency between said instruction and said load operations.

7. The apparatus of claim 1 further comprising a cache controller to control accesses to cache memory made in response to said execution unit executing a load operation.

8. The apparatus of claim 7 wherein said cache controller comprises a control register to control serialization of cache memory accesses made in response to executing said load operation.

9. A processor comprising:

a decoder circuit to decode a load fence instruction;

a control register to enable a pre-serialization mode, a post-serialization mode, and an enhanced mode for said load fence instruction;

an execution unit to execute said load fence instruction after said decode circuit has decoded said load fence instruction.

10. The processor of claim 9 wherein said execution unit prevents load operations from executing until load operations executed prior to said load fence instruction are globally observed.

11. The processor of claim 10 wherein said execution unit blocks an instruction subsequent to said load fence instruction until said execution unit executes said load fence instruction.

12. The processor of claim 11 wherein said execution unit allows an instruction subsequent to said load fence instruction to execute out of order with respect to said load fence instruction if there is no dependency between said instruction and said load operations.

13. The processor of claim 12 further comprising a cache controller to control accesses to cache memory made in response to said execution unit executing a load operation.

14. The processor of claim 13 wherein said cache controller comprises a control register to control serialization of cache memory accesses made in response to executing said load operation.

15. A method comprising:

decoding a load fence instruction;

executing said load fence instruction after said decode circuit has decoded said load fence instruction.

16. The method of claim 15 wherein said execution unit prevents load operations from executing until load operations executed prior to said load fence instruction are globally observed.

17. The method of claim 16 wherein a load buffer stores load operations to be executed by said execution unit.

18. The method of claim 17 wherein a reorder buffer stores load data resulting from executing said load operations.

19. The method of claim 18 wherein said execution unit blocks an instruction subsequent to said load fence instruction until said execution unit executes said load fence instruction.

20. The method of claim 19 wherein said execution unit allows an instruction subsequent to said load fence instruction to execute out of order with respect to said load fence instruction if there is no dependency between said instruction and said load operations.

21. The method of claim 20 further comprising a cache controller to control accesses to cache memory made in response to said execution unit executing a load operation.

22. The method of claim 21 wherein said cache controller comprises a control register to control serialization of cache memory accesses made in response to executing said load operation.